Low-Power 45nm 1Mb SRAM

Mark Cheung, Austin Moran, Xiafei Yang
ECE 4332 – Fall 2013
University of Virginia
< mc5ah, apm4yw, xy2wf>@virginia.edu

ABSTRACT

A solution methodology of designing a 1MB low-power SRAM is developed to meet Portable Instruments Company's specifications. The primary goal is to minimize the total power with reasonable sizing and delay, which means focusing on the optimization of dimensions of array blocks of the entire memory array and the structure of its periphery circuits to access the bitcells as well as source voltage. Our design is well functioning at all process corners, and a range of supply voltages and temperatures. The final metric of our SRAM is 4.96*10^-36 J^2*s *mm*W

1. INTRODUCTION

Portable Instruments Company (PICo) is seeking for a 1Mb low power SRAM design for their application of a microsensor node. This application is required to have a long lifetime, which means that the major constraint is on energy consumption. The key metric to optimize energy consumption is (Active Energy per Access)^2*Delay*Area*Idle Power. And the memory design must be robust across variations of process, voltage and temperature corners. To meet PICo's design specifications, our team has designed a well functional SRAM using 45nm FreePDK technology and Cadence software.

2. SRAM OVERVIEW

A typical structure for an SRAM contains decoders, memory array, sense amplifier and periphery circuit to access the bit cells. There are three different components of energy dissipation in an SRAM [5], 1.) the dynamic energy to switch the capacitance in the decoders, bitlines, datalines and other control signals within the array; 2.) the energy of the sense amplifiers; 3.) the energy loss due to the leakage currents. In order to significantly reduce the power, we look at the following 1) capacitance reduction of wordlines, bitlines and decoders; 2.) current reduction for wordlines, periphery circuits and sense amplifier; 3.) operating voltage reduction. The large capacitive elements in a memory are wordline, bitlines and datalines each with a number of cells connected to them. So reducing the size of these lines can have a great impact on capacitance reduction and therefore save the dynamic energy to switch capacitance. [7] For this purpose, the

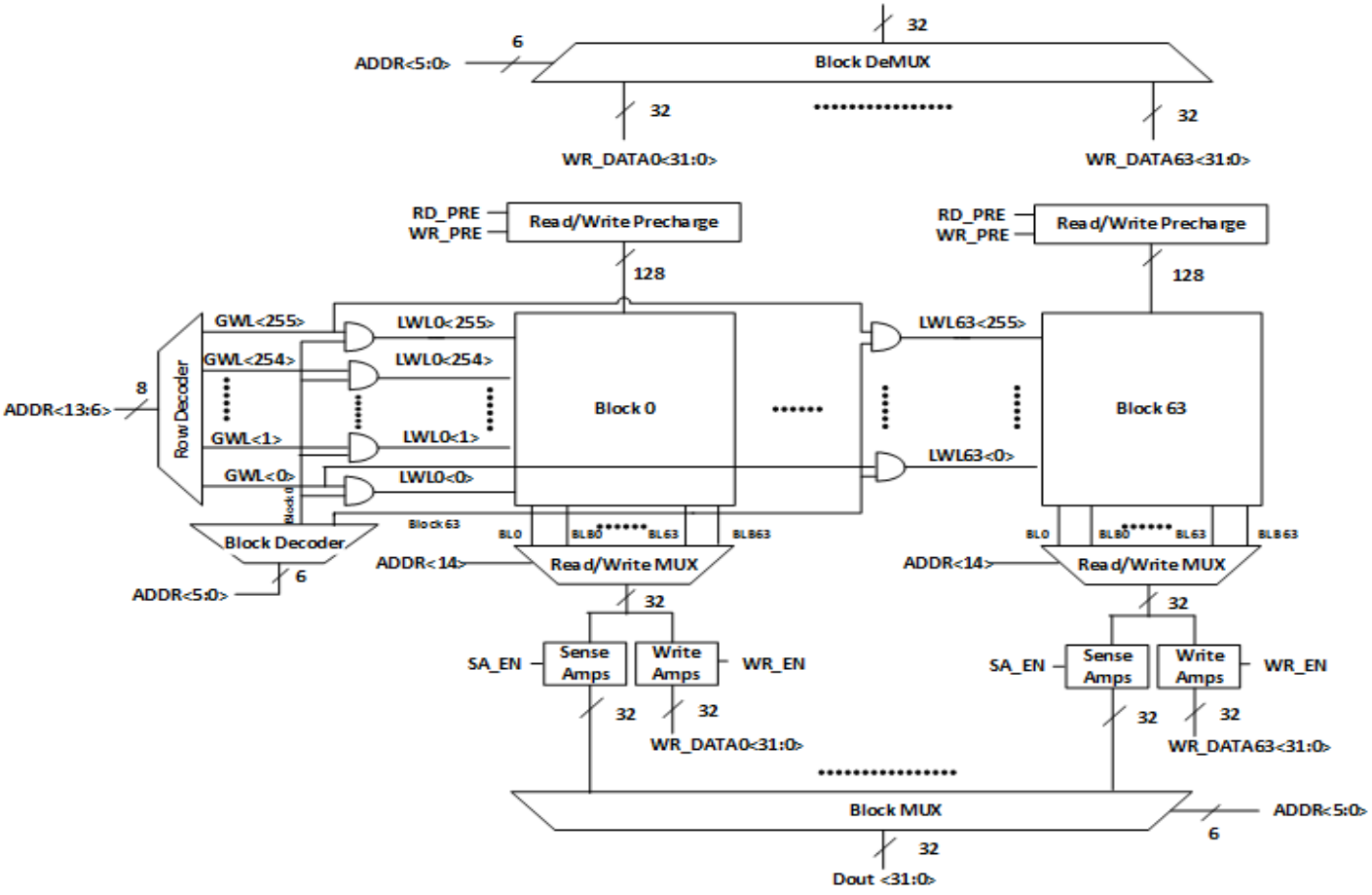

Figure 1. Global block diagram

decoder will employ the divided word line structure, where part of the address is decoded to access the horizontal global word line and the remaining address bits activate the vertical block select line. The intersection of these two activates the local word line. The cells connected to this word line transfer their data onto the bit lines. Data from a subset of bit lines is routed by the column mux into the sense amplifiers which amplify and drive it onto the data lines.[5] Figure 1 is the SRAM model that shows the above arrangements.

3. BITCELL

A 1Mb SRAM is made up of a little over a million bitcells. The sizing of the bitcells' transistors correlate directly with area. The standard 6T bitcell was chosen for its stability, area, and energy (lower static power loss compared to 4T bitcell with resistors). Using 90nm as the minimum width, we chose 90nm, 135nm, and 180nm widths respectively for the pull-up, passgate, and pull-down transistors. These widths give us better cell ratio and pull ratio, and hence better performance (lower failure probability estimated by Monte Carlo).

3.1 Bitcell Layout

The layout [2] we used differs greatly from the standard industry layout in that the gates are now in line with the inputs to the cell. Compared to the industry standard, we found the layout to have a lower capacitance at the trade-off of higher area. The energy required to charge the bit line is reduced (due to its proportionality to the effective capacitance) and allow the sense amp to activate earlier (lower capacitance results in a lower time constant = BL,BLB change faster).

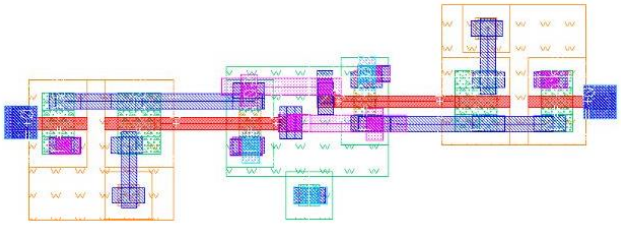

Figure 2. Layout of a single bitcell using the propose method in [2]

We simulated the static noise margins for bitcells at 0.7 Vdd at all process corners. All of which succeed, and the worst SNM is found to be FS corners for both read and hold. We use this worse case corner for the metrics computation.

3.2 Bitcell Static Noise Margin(SNM)

We have simulated our bitcell to make sure the SNM succeed in every .process corners. Butterfly curves for the worst case corners are shown in figure 3 and 4. Figure 5 s

Corners	Read(volts)	Hold(volts)
FS	0.121	0.241
SS	0.212	0.297
FF	0.205	0.299
TT	0.203	0.294
SF	0.210	0.228

Table 1. SNM values from five process corners for both read/hold

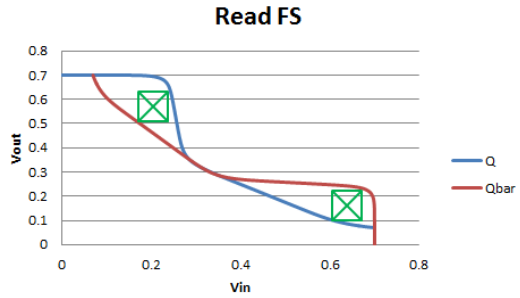

Figure 3. Worst case read corner

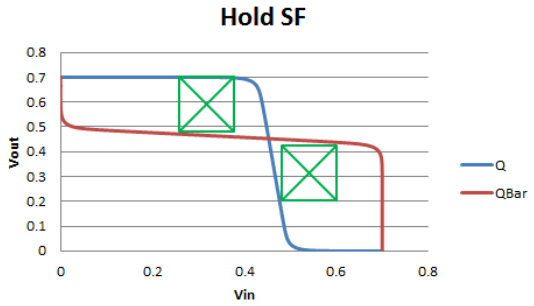

Figure 4. Worst case hold corner

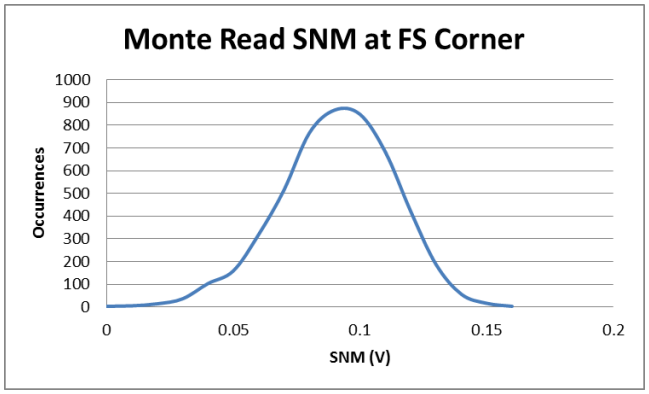

Figure 5. Read SNM at worst case corner with 5000 iterations

4. SRAM ARRAY MODEL

To optimize our SRAM architecture, we have built a SRAM model to get into details of each major components with all the peripheral circuits and simulate the read and write processes. The model includes blocks of memory array, address decoders to access the selected word, bitlines, word lines, read/write MUX, sense and write amplifiers and block MUX and deMUX.

4.1 Blocks

As our array model accesses one word, all the other words are in active. The block model hence contains bitcells of multiplicity 1. Four extra bitcells approximate the load on the word lines and bitlines and the leakage of the other bitcells. Their multiplicities depend on the number of rows (r) and columns (c). An additional block model was used to model the leakage in all the other blocks. Since the bitlines are long, an RC load was added to model the parasitic characteristic of the wires. The resistance and

capacitance values were obtained from the NCSUwiki [3] and parasitic extraction, respectively.

The sizing for our blocks was based on previous groups work. Our values of (r =256, c =64, b =64) were chosen to provide a close fit to the optimal energy delay curve. The values used do not provide the best fit, but allow for some delay slack due to circuitry outside of the bitcells [4]. Splitting the memory array into multiple blocks also reduces the sizing and thus energy usage of the decoders.

4.2 Decoders/Predecoder

The decoder section consists of the local word line drivers (LWL), the global word line drivers (GWL) and the block select (BS). Delay is key to optimizing the decoders. The longer it takes the decoders to select a LWL the longer the clock period must be. However focusing on only speed has the downside of using more energy. Accordingly the decoders were laid out and sized for a balance of speed and energy usage. This was done according to heuristic H3 which is laid out in [6]. H3 recommends to size the input gates to each section minimally and to then size each section independently (LWL, GWL, and BS) according to their respective loads. Both the GWL and BS consist of two sections: a predecoder and a decoder section. The decoder section is composed of two input NAND gates with inverters and the predecoder is implemented with either four or three (GWL, BS) input NAND gates with inverters. The decoder section was kept to two input NAND gates to minimize power dissipation and larger input NAND gates were used for the predecoders for speed (as it minimizes the number of levels that the decoder needs to have). The LWL drivers are two input NAND gates with inverters sized to drive the resistive-capacitive load of the local word line and access transistors of the bitcells.

4.3 Word Lines

The word lines were modeled as a repeated L-network of capacitors connected to ground and the LWL and resistors between each capacitor. This is done to model the wire resistance and capacitance inherent to the word lines as well as the gate capacitance of the access transistors that load the LWL.

4.4 Bitlines

As well the bitlines were modeled as a repeated L-network of capacitors and resistors. This is to model the wire resistance and capacitance as well as the capacitance that each bitcell loading the bit lines adds.

4.5 Read/Write MUXes

The read/write (R/W) multiplexers are implemented as pairs of transmission gates (one T-Gate pair for each bit line pair). The mulitplexers exist to select which word in the row is read or written. Transmission gates were chosen for their low energy usage and their ability to pass signals through in both directions. This is important as it allows the sense amplifiers to read the discharging bit lines and allows the write amps to pull down the bit lines according to the data being written.

4.6 Sense Amps

For the sense amplifiers we used a voltage-sensing, latching amplifier (figure 5). This design was chosen for its latching ability, speed, and lack of effect on bit lines during a read. The latching ability means that once the amplifier has sensed a voltage difference the output is held constant for as long as the sense amp enable signal stays high. The latched output means that we can allow the bit lines to drop, fire the sense amp, stop the bit lines from discharging, and still keep the output valid. This should prevent excessive discharge of the bit lines and save energy per read as a result. As far as speed goes, the amplifier is capable of sensing a minimum voltage difference of 40mV. A smaller voltage difference requirement means that the bit lines have to discharge less and thus should save energy and delay. By lack of effect on the bitlines: the sense amplifier inputs are the gates of two NMOS transistors. Gates of MOSFETs are usually treated as capacitors so they shouldn't draw any current from the bit lines during a read.

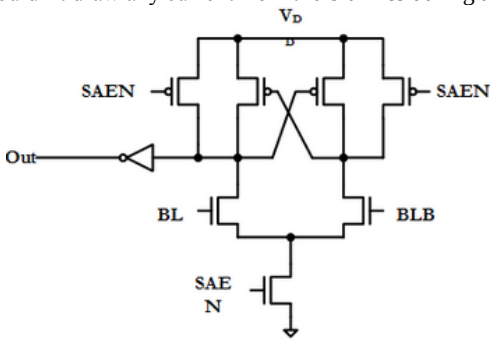

Figure 5: Sense amp model

4.7 Write Amps

A single write amplifier consists of a two large NMOS transistor with the drains connected to the R/W Mux, the source connected to the data line, and the gate connected to the word select bit. The write amplifiers are structured into 64 pairs of NMOS transistors; a pair of transistors for each bit line pair and 32 pairs per word. The size of the write amps is important as they are responsible for pulling down the bitlines during a write. They need to be large enough to pull down the highly capacitive bitline quickly. To make sure the bitline and bitlinebar receive different values, there is an inverter between the sources of the bitline write amp and bitlinebar write amp.

4.8 Block MUX/DeMUX

The Block Mux and deMux would have been implemented using transmission gates for minimal power usage. The block mux serves to choose which block's sense amplifier data is sent to the final 32 bit output. The block demux works to send the write data to the correct block. Both prevent having to send the same signal to every block which should reduce the fanout of associated gates, reducing the delay of the signal inputs.

5. SPECIAL FEATURE

In addition to the standard functionality of an SRAM, our team has also explored some special features to modify our design for an optimal low-power SRAM.

5.1 High threshold voltage

6T transistors used in bitcell are all high VT devices. Simulations performed within the bitcell showed significant reduction in leakage power and energy at the trade-offs of delays using different VT devices at VDD of 0.7 V. We calculated a quick

metrics that is similar to our main metrics (since area is the same and max write power should just be taken over average). From the table below, we can see that VTH transistors are the most preferable.

Threshold device	Idle Power	Max write power ~E	Write delay ~D	M= P*E^2*D
VTL	35.2uW	62.3uW	.067ns	9261.7
VTG	4.29uW	44.87nW	.75ns	679.49
VTH	.054W	30.2uW	.17ns	8.84

Table 2. Analysis of different VT devices corresponding to the key metric

5.2 Optimal VDD

After finding the minimum VDD with reasonable hold and read SNM, we sweep it to a slightly higher VDDs and found that the decrease in delay is not sufficient to offset the increase in energy and power. Hence, we use 0.7 V for our final VDD.

5.3 Predecoding

Please refer to 4.2 section for more detail.

6. RESULTS

Based on the key evaluation metric that PICo provides, we have calculated our design in all the aspects.

6.1 Energy

PICo defined energy as the average of 5 reads and 1 write. We calculated the average current going through the bitline/bitlineb during both accesses and multiply by 64 (cells/row), VDD, and time.

6.2 Delay

Delay for worst case read and write is calculated at 0.63 V FS process corner (worst corner). Each delay is calculated using the array model plus the delay of the decoder (1.25n). For a read, minimum of 40mV differential is needed between the bitlines. Hence, reading took longer. Minimum write signal is found by decreasing the write pulse until it fails. It is significantly lower (see below figure).

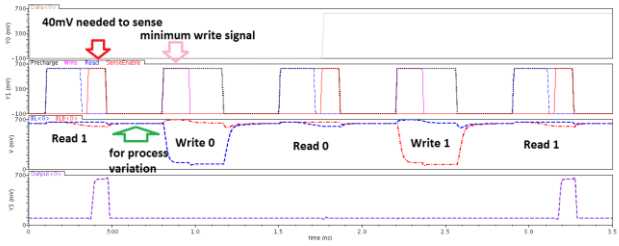

Figure 6. Timing diagram of 3 reads and 2 writes at FS 0.63 V process corner.

6.3 Area

PICo defined area as a rectangle box that completely encompasses the SRAM layout. The layout, created using skill, is not entirely complete, but the majority is expected to be consumed by the 1M bitcells, which accounts for 1.82 mm^2. Hence the total area is roughly 2.2 mm^2 (1.82 mm^2 for all the bitcells + peripherals).

6.4 Idle power

PICo defined idle power as the power when the array is not being accessed. We found this by adding the idle power of a bitcell (times 2^20) and the idle power of the peripherals using average current times voltage.

6.5 Final metric

Metric	Value
1 Bitcell Area	1.69um^2
Total Area	2.2 mm^2
Read Energy	2.9 pJ
Write Energy	3.1 pJ
Total energy	2.93 pJ
Read Delay	2.6ns
Write Delay	1.6ns
Total Delay	2.6ns
Idle Power	101 uW
Total Metric	4.96*10^-36 J^2*s *mm*W

Table 3. Final Metric Calculation

7. CONCLUSION

We have demonstrated a functional low-power 1Mb memory with 32-bit word. To optimize energy, we use high voltage threshold transistors, lower the VDD to 700mV, utilized a lower bitline capacitance layout, predecoding, and perform monte carlo testing to assure the functionality of read and write accesses. Our metrics is almost as good as the 2011 group. There are two main differences. The first is the area: we increased the widths of all the transistors to decrease the Monte Carlo failure rates. The second difference is the power: our idle power (due to leakage) is significantly lower as we use high voltage threshold transistors.

Future extension includes adding error correcting curves, and other knobs to move further on the pareto curves: lengths, multiple VDD, data retention gate-grounded.

8. ACKNOWLEDGMENTS

Our team would like to thank Professor Benton Calhoun for providing resourceful feedbacks about PICo's design specifications, and Professor Aatmesh Shrivastava for technical assistance of our project. We would also like to thank Divya Akella, Jim Boley, and Alicia Klinefelter for their advice and help on SRAM design and FreePDK technology implementation, and teams from previous years, along with our fellow team--VLSE for sharing their experience.

9. REFERENCES

[1] Jacob, B., Ng, S., & Wang, D. (2008). *Memory systems : cache, DRAM, disk.* San Francisco, California: Morgan Kaufmann .

[2] Mann, R., Calhoun, B. (2011). *New category of ultra-thin notchless 6T SRAM cell layout topologies for sub-22nm* . 12th International Symposium on Quality Electronic Design

[3] Itoh, K., Horiguchi, M., Tanaka, H. (2007) *Ultra-low voltage nano-scale memories.* New york, New york: Springer Science+Business Media

[4] Bailey, S., Linger, K., Lorenzo, R., & Thompson, J. (2011). Team 1 implementation of a low power SRAM design using 45 nm FreePDK technology. Unpublished.

[5] Bharadwaj S. Amrutur and Mark A. Horowitz.(2000) *Speed and Power Scaling of SRAM's.* IEEE TRANSACTIONS ON SOLID-STATE CIRCUITS, VOL. 35, NO. 2, FEBRUARY 2000

[6] B. S. Amrutur (1999) *Design and Analysis of Fast Low Power SRAMs.* A Dissertation submitted to the department of Electrical Engineering and the Committee on Graduate Studies of Stanford University in partial fulfillment of the requirements for the degree of Doctor of Philosophy

[7] Martin Margala. (1999) *Low-Power SRAM Circuit Design.* Records of the IEEE International Workshop on Memory Technology, Design and Testing, August 9-10, 1999 San Jose, California

[8] Der-Chen Huang. (No Date Listed) *Sense Amplifier for SRAM.* http://soc.cs.nchu.edu.tw/upload_data/Sense%20Amplifier%20for%20SRAM.pdf

[9] J. S. Caravella, *A 0.9V, 4K SRAM for Embedded Applications.* in Proceedings of CICC, pp.119-122, May 1996

[10] J. S. Caravella, *A Low Voltage SRAM For Embedded Applications*. IEEE Journal of Solid-State Circuits, vol. 32, no. 3, pp. 428-432, March 1997

[11] Cabe, A. (2006) ToolsSimulationMemoryStaticNoiseMargin https://venividiwiki.ee.virginia.edu/mediawiki/index.php/ToolsSimulationMemoryStaticNoiseMargin

[12] Chen, Y., Converse, C., Gan, C., & Moore, D. (2010) ClassECE4332Fall10ProjectTeam2 https://venividiwiki.ee.virginia.edu/mediawiki/index.php/ClassECE4332Fall10ProjectTeam2

[13]Wang, J., Lee, H. (1998) "A new current-mode sense amplifier for low- voltage low-power SRAM design", Eleventh Annual IEEE International Proceeding of ASIC, pp.163-167, Sep. 1998

[14] Blalock, T.N., Jaeger, R.C. (1991) "A High-speed Clamped Bit-line Current-mode Sense Amplifier", IEEE J. Solid-State Circuits, vol. 26, no. 4, pp542-548, April 1991